IN THE CLAIMS

The following is a listing of the claims of the present application:

1. (Currently Amended) A method of generating a web-based application, the method comprising the steps of:

composing one or more web pages in accordance with a scripting language to form the application; ~~and~~

embedding one or more extended function calls in the one or more web pages in accordance with the scripting language such that the application, when executed by a computer system in which the application is installed, has access to one or more operating system resources of the computer system that are not associated with a context of a web browser through the one or more extended function calls;

wherein the one or more embedded extended function calls cause one or more application programming interfaces of an operating system to be executed in order to access the one or more operating system resources that are not associated with a context of a web browser; and

providing an application manager that manages a life-cycle associated with the web-based application in accordance with the computer system, wherein the application manager is operative to: (i) process code in each web page of the application; (ii) invoke the web browser to process code that is of a visual presentation type; (iii) invoke a data modeling language parser to parse code that is of a data modeling language type; (iv) invoke a scripting language interpreter to parse code that is of the scripting language type such that the scripting language interpreter may execute code that is of the original scripting language type used by the web browser; and (v) invoke an operating system interface module to execute code, successfully parsed by the scripting language interpreter, that executes the one or more application programming interfaces.

2. (Canceled).

3. (Previously Presented) The method of claim 1, wherein the scripting language includes code for accessing one or more operating system resources of the computer system that are associated with a context of a web browser.

4. (Canceled).

5. (Canceled).

6. (Currently Amended) A software system for processing a web-based application in accordance with a computer system, the software system comprising:

an application manager that manages a life-cycle associated with the web-based application in accordance with the computer system, wherein the application is composed of one or more web pages and has access to one or more operating system resources of the computer system that are not associated with a context of a web browser;

a scripting language interpreter that interprets scripting language associated with the one or more web pages of the application; and

an operating system interface module which converts one or more calls embedded in the interpreted scripting language associated with the one or more web pages into code that executes one or more application programming interfaces so as to access the one or more operating system resources of the computer system that are not associated with a context of a web browser;

wherein the application manager is operative to: (i) process code in each web page of the application; (ii) invoke the web browser to process code that is of a visual presentation type; (iii) invoke a data modeling language parser to parse code that is of a data modeling language type; (iv) invoke the scripting language interpreter to parse code that is of the scripting language type such that the scripting language interpreter may execute code that is of the original scripting language type used by the web browser; and (v) invoke the operating system interface module to execute code, successfully parsed by the scripting language interpreter, that executes one or more application programming interfaces.

7. (Original) The software system of claim 6, further comprising a web browser to at least one of retrieve web objects, send web requests, and provide a graphical user interface in accordance with the execution of the web-based application.

8. (Original) The software system of claim 6, further comprising a data modeling language parser to decode information in the web-based application written in a corresponding data modeling language.

9. (Previously Presented) The software system of claim 8, wherein the data modeling language parser is an XML parser.

10. (Canceled).

11. (Previously Presented) The software system of claim 6, wherein the scripting language is JavaScript™.

12. (Previously Presented) The software system of claim 6, wherein the scripting language interpreter is a JavaScript™ interpreter.

13. through 47. (Canceled).

48. (Previously Presented) A software system for processing a web-based application in accordance with a computer system, the software system comprising:

an application manager that manages a life-cycle associated with the web-based application in accordance with the computer system, wherein the application is composed of one or more web pages and has access to one or more operating system resources of the computer system that are not associated with a context of a web browser;

a scripting language interpreter that interprets scripting language associated with the one or more web pages of the application;

an operating system interface module which converts one or more calls embedded in the interpreted scripting language associated with the one or more web pages into code that executes one or more application programming interfaces so as to access the one or more operating system resources of the computer system that are not associated with a context of a web browser, wherein

the one or more operating system resources of the computer system that are not associated with a context of a web browser comprise a screen display outside of a window of the web browser;

a web browser; and

a data modeling language parser;

wherein the application manager is operative to: (i) process code in each web page of the application; (ii) invoke the web browser to process code that is of a visual presentation type; (iii) invoke the data modeling language parser to parse code that is of a data modeling language type; (iv) invoke the scripting language interpreter to parse code that is of the scripting language type such that the scripting language interpreter may execute code that is of the original scripting language type used by the web browser; and (v) invoke the operating system interface module to execute code, successfully parsed by the scripting language interpreter, that executes the one or more application programming interfaces.

49. (Previously Presented) A software system for processing a web-based application in accordance with a computer system, the software system comprising:

an application manager that manages a life-cycle associated with the web-based application in accordance with the computer system, wherein the application is composed of one or more web pages and has access to one or more operating system resources of the computer system that are not associated with a context of a web browser;

a scripting language interpreter that interprets scripting language associated with the one or more web pages of the application;

an operating system interface module which converts one or more calls embedded in the interpreted scripting language associated with the one or more web pages into code that executes one or more application programming interfaces so as to access the one or more operating system resources of the computer system that are not associated with a context of a web browser, wherein the one or more operating system resources of the computer system that are not associated with a context of a web browser comprise a file management system;

a web browser; and

a data modeling language parser;

wherein the application manager is operative to: (i) process code in each web page of the application; (ii) invoke the web browser to process code that is of a visual presentation type; (iii) invoke the data modeling language parser to parse code that is of a data modeling language type; (iv) invoke the scripting language interpreter to parse code that is of the scripting language type such that the scripting language interpreter may execute code that is of the original scripting language type used by the web browser; and (v) invoke the operating system interface module to execute code, successfully parsed by the scripting language interpreter, that executes the one or more application programming interfaces.